



# Studienarbeit - Entwurf

Ausarbeitung InfluxDB und Wetterdaten

Erstellt von:

Henrik Mertens  
Hatzfelder Str 25  
33104 Paderborn

Prüfer:

Prof. Dr. Ulrich. Reus

Eingereicht am:

29. Mai 2022

# Inhaltsverzeichnis

Abkürzungsverzeichnis	IV
Abbildungsverzeichnis	V
Tabellenverzeichnis	VI
Listingverzeichnis	VII
<b>1 Einleitung</b>	<b>1</b>
1.1 Zielsetzung . . . . .	1
1.2 Aufbau und Vorgehensweise . . . . .	1
<b>2 Grundlagen</b>	<b>2</b>
2.1 Time Series Data . . . . .	2
2.2 Unterschiede zwischen Time Series und relationalen Datenbanken . . . . .	3
2.3 Verbreitete DBMS . . . . .	3
2.4 Entwicklungsumgebung . . . . .	5
2.4.1 Docker . . . . .	5
2.4.2 Python und Jupyter Notebooks . . . . .	5
2.4.3 InfluxDB installation . . . . .	5
<b>3 InfluxDB</b>	<b>6</b>
3.1 Daten einfügen . . . . .	6
3.2 Daten abrufen . . . . .	7
3.3 Daten verarbeiten und Visualisieren . . . . .	7
3.4 weitere InfluxDB-Funktionen . . . . .	7
3.5 Wetterdaten . . . . .	7
3.5.1 Wetterdaten Aufbau . . . . .	7
3.5.2 Wetterdaten abrufen . . . . .	9
3.5.3 Wetterdaten verarbeiten . . . . .	9
<b>4 Zusammenfassung</b>	<b>10</b>
<b>Anhang</b>	<b>11</b>
<b>Quellenverzeichnis</b>	<b>13</b>

**Ehrenwörtliche Erklärung**

**14**

## Abkürzungsverzeichnis

API	Application Programming Interface.
CLI	Command Line Interface.
CRUD	Create, Read, Update, Delete.
DWD	Deutscher Wetterdienst.
HTTP	HyperText Transfer Protocol.
IOT	Internet of Things.
RDBMS	Relational Database Management System.
TSDB	Time Series Database.
UI	User Interface.

## Abbildungsverzeichnis

Abbildung 1: DB-Engines Ranking . . . . .	4
---	---

## Tabellenverzeichnis

## Listingverzeichnis

Listing 1: InfluxDB Line Protokoll . . . . .	7
Listing 2: Wetterdaten CSV . . . . .	8

# 1 Einleitung

## 1.1 Zielsetzung

Das Ziel dieser Arbeit ist es eine Einführung in die Funktion von Time Series Database (TSDB) zu geben. Außerdem soll beispielhaft an InfluxDB gezeigt werden wie mit einer TSDB gearbeitet wird. Dazu werden die Wetterdaten vom Deutschen Wetterdienst (DWD) Importiert und ausgewertet.

## 1.2 Aufbau und Vorgehensweise

Im ersten Teil dieser Arbeit werden die Grundlagen von TSDB erklärt und Besonderheiten beschreiben. Im darauf Folgenden Kapitel wird dann exemplarisch an InfluxDB gezeigt wie mit einer TSDB gearbeitet wird.

Im letzten Kapitel werden die Inhalte dieser Arbeit zusammengefasst.



## 2 Grundlagen

TSDB gehören zu den NoSQL Datenbanken und sind besonders darauf optimiert mit Time Series Data zu arbeiten. Daruch können die große Mengen an Time Series Daten verarbeiten durchsuchen und Speichern.<sup>1</sup>

### 2.1 Time Series Data

Um TSDB zu verstehen muss als erstes geklärt werden was Time Series Data überhaupt ist und wie sie sich von anderen Daten unterscheiden. Wie der Name schon sagt ist Time Series Data eine Reihe von Daten die über einen Zeitraum gesammelt wordern sind. Es wird also nicht nur der Endwert aufgezeichnet sonder die Veränderung über einen Zeitraum. Diese Daten können z.B. Servermetriken, Anwendungs Leistungsüberwachung, Netzwerkdaten, Internet of Things (IOT) Sensordaten, Ereignisse, Klicks, Marktgeschäfte und viele andere Arten von Daten sein. Time Series Data können gut daran erkannt, dass die Zeit eine Wichtige Axe bei der Darstellung der Werte ist.<sup>2</sup>

Manchmal ist es nicht notwendig alle Daten zu erfassen. Zum Beispiel wird in vielen Anwendungen nur der letzte Login gespeichert und mehr ist auch für die Funktion nicht notwendig. Allerdings können zusätzliche Informationen gewonnen werden wenn nicht nur die letzten Daten sondern die Veränderung aufgezeichnet werden. So kann zum Beispiel festgestellt werden wie oft und wann sich der Kunde einloggt und ob es dabei ein Muster gibt. Anhand dieser Daten können Kunden dann Kategorisiert werden.<sup>3</sup>

Eine Besonderheit von Time Series Data ist das sie sich nicht verändert. Wenn die Daten einmal erfasst wurden wird an ihnen nichts mehr verändert. Es werden nur neue Daten hinzugefügt<sup>4</sup>

---

<sup>1</sup>vgl. ComputerWeekly.de, Redaktion (2021)

<sup>2</sup>vgl. Dix, Paul (2021), S. 1 ff.

<sup>3</sup>vgl. Data-Science-Team (2020)

<sup>4</sup>vgl. Fangman, Sam (2019)

## 2.2 Unterschiede zwischen Time Series und relationalen Datenbanken

Um Time Series Data zu speichern ist es nicht unbedingt erforderlich eine TSDB zu nutzen. Auch relationalen Datenbanken können Time Series Data speichern. Einer der wichtigsten Unterschiede zwischen einer TSDB im Gegensatz zu einer Relational Database Management System (RDBMS) ist es, dass kein Datenbank Schema notwendig ist. Wenn Time Series Daten in eine Relationale Datenbank geschrieben werden sollen müssen erst entsprechende Tabellen angelegt werden in denen die Daten immer im gleichen Format abgelegt werden müssen. Im Gegensatz dazu können in einer TSDB die Daten einfach Schemafrei in die Datenbank geschrieben werden. Ein weiterer Vorteil ist es, dass TSDB im Gegensatz zu relationalen Datenbanken besser und einfacher skaliert werden können.<sup>5</sup>

Aber TSDB haben nicht nur Vorteile. Wie in Abb. 1 zu sehen sind sie viel weniger verbreitet als Zeitbasierte Datenbank Systeme. Dadurch gibt es viel weniger Entwickler die sich mit TSDB auskennen und auch das Ökosystem um die Datenbank ist deutlich kleiner. Außerdem sind RDBMS dadurch dass es sie viel länger gibt sehr stabil und sehr gut unterstützt.<sup>6</sup>

RDBMS arbeiten nach dem Create, Read, Update, Delete (CRUD) Prinzip welches für Time Series Data nicht optimal ist. Auf Time Series Data werden keine Update Befehle durchgeführt, da neue Daten immer nur als neuer Datenpunkt angehängt werden. Auch das Löschen von Daten wird nicht sehr häufig durchgeführt und im Gegensatz zu RDBMS meistens gleichzeitig auf einer großen Menge an Datensätzen. Daher sind TSDB besser dafür geeignet mit Time Series Data zu arbeiten und weisen auch eine höhere Performance auf.<sup>7</sup>

## 2.3 Verbreitete DBMS

Aktuell gibt es wie in Abb. 1 zu sehen einige beliebte Multi-Model Datenbanken die als TSDB genutzt werden können. So können die Datenbanken MongoDB, Redis, Teradata

---

<sup>5</sup>vgl. Influxdata (2021)

<sup>6</sup>vgl. Influxdata (2021)

<sup>7</sup>vgl. Influxdata (2021)

und Couchbase mit Time Series Daten arbeiten. Die erste reine TSDB im Ranking ist InfluxDB auf Platz 29.<sup>8</sup>

Abbildung 1: DB-Engines Ranking

394 Systeme im Ranking, Mai 2022

Rang			DBMS	Datenbankmodell	Punkte		
Mai 2022	Apr 2022	Mai 2021			Mai 2022	Apr 2022	Mai 2021
1.	1.	1.	Oracle +	Relational, Multi-Model	1262,82	+8,00	-7,12
2.	2.	2.	MySQL +	Relational, Multi-Model	1202,10	-2,06	-34,28
3.	3.	3.	Microsoft SQL Server +	Relational, Multi-Model	941,20	+2,74	-51,46
4.	4.	4.	PostgreSQL +	Relational, Multi-Model	615,29	+0,83	+56,04
5.	5.	5.	MongoDB +	Document, Multi-Model	478,24	-5,14	-2,78
6.	6.	7.	Redis +	Key-value, Multi-Model	179,02	+1,41	+16,85
7.	8.	6.	IBM Db2	Relational, Multi-Model	160,32	-0,13	-6,34
8.	7.	8.	Elasticsearch +	Suchmaschine, Multi-Model	157,69	-3,14	+2,34
9.	9.	10.	Microsoft Access	Relational	143,44	+0,66	+28,04
10.	10.	9.	SQLite +	Relational	134,73	+1,94	+8,04
11.	11.	11.	Cassandra +	Wide column	118,01	-3,98	+7,08
12.	12.	12.	MariaDB +	Relational, Multi-Model	111,13	+0,81	+14,44
13.	13.	13.	Splunk	Suchmaschine	96,35	+1,11	+4,24
14.	14.	27.	Snowflake +	Relational	93,51	+4,06	+63,46
15.	15.	15.	Microsoft Azure SQL Database	Relational, Multi-Model	85,33	-0,45	+14,88
16.	16.	16.	Amazon DynamoDB +	Multi-Model	84,46	+1,55	+14,39
17.	17.	14.	Hive +	Relational	81,61	+0,18	+5,42
18.	18.	17.	Teradata +	Relational, Multi-Model	68,39	+0,82	-1,59
19.	19.	19.	Neo4j +	Graph	60,14	+0,62	+7,91
20.	20.	20.	Solr	Suchmaschine, Multi-Model	57,26	-0,48	+6,07
21.	21.	18.	SAP HANA +	Relational, Multi-Model	55,09	-0,71	+2,33
22.	22.	22.	FileMaker	Relational	52,27	-0,64	+5,55
23.	24.	24.	Google BigQuery +	Relational	48,61	+0,63	+10,98
24.			Databricks	Multi-Model	47,85		
25.	23.	21.	SAP Adaptive Server	Relational, Multi-Model	47,78	-0,58	-2,19
26.	25.	23.	HBase +	Wide column	43,19	-1,14	-0,05
27.	26.	25.	Microsoft Azure Cosmos DB +	Multi-Model	40,22	-0,12	+5,51
28.	27.	28.	PostGIS	Spatial DBMS, Multi-Model	31,82	-0,23	+1,98
29.	28.	29.	InfluxDB +	Time Series, Multi-Model	29,55	-0,47	+2,38
30.	29.	26.	Couchbase +	Document, Multi-Model	28,38	-0,67	-1,85

Quelle: <https://db-engines.com/de/ranking?msckid=4f2a29e5d08811ec95ccd74f8f5146ab>

Allerdings haben Datenbanken die nur auf das Verarbeiten von Time Series Data ausgelegt sind deutliche Performance Vorteile gegenüber Multi Model Datenbanken. In einem Vergleich von InfluxDB und MongoDB hat InfluxDB eine 2,4 mal bessere Schreibperformance als MongoDB und ist beim Lesen sogar 5,7 mal schneller. InfluxDB benötigt außerdem 20 mal weniger Speicherplatz auf der Festplatte um die gleiche Menge an Daten zu speichern.<sup>9</sup>

<sup>8</sup>vgl. solid-IT-gmbh (2022)

<sup>9</sup>vgl. Hajek Vlasta Pour Ales, Kudibal Ivan (2019)

## **2.4 Entwicklungsumgebung**

### **2.4.1 Docker**

### **2.4.2 Python und Jupyter Notebooks**

### **2.4.3 InfluxDB installation**

## 3 InfluxDB

InfluxDB ist eine in Go geschriebene open source TSDB die darauf ausgelegt ist mit großen Mengen an Time Series Data zu arbeiten.<sup>10</sup> In Kapitel 4 dieser Arbeit wird am Beispiel von Wetterdaten gezeigt wie mit InfluxDB gearbeitet wird. InfluxDB stellt für die Integration in eigene Anwendungen ein HyperText Transfer Protocol (HTTP) Application Programming Interface (API) zur Verfügung für die es in vielen Programmiersprachen Client Libraries gibt. Außerdem wird ein Webinterface und ein Command Line Interface (CLI) bereitgestellt.<sup>11</sup>

### 3.1 Daten einfügen

In InfluxDB werden Daten immer in Buckets gespeichert. Um Daten hochzuladen muss zunächst erst einmal ein Bucket angelegt werden. Dazu gibt es zwei Möglichkeiten. Die einfachste ist es über das Web User Interface (UI) von InfluxDB einen neuen Bucket anzulegen. Dazu muss nach dem Login der Navigationspunkt Data und der Reiter Buckets ausgewählt werden. Hier kann dann mit dem Button Create Bucket ein neuer Bucket angelegt werden. Beim Anlegen kann noch eine Lebensdauer für die Daten ausgewählt werden nach welcher die jeweiligen Datenpunkte gelöscht werden.

Daten werden immer nach dem InfluxDB Line Protokoll formatiert an die Datenbank gesendet. Das Protokoll ist wie in Listing 1 dargestellt aufgebaut. Im ersten Teil des Line Protokolls wird der Name der Messreihe angegeben. Das kann zum Beispiel der Name des Sensors sein oder der Ort an dem der Messwert genommen wurde. Wichtig ist, dass Groß- und Kleinschreibung beachtet werden muss und Unterstriche nicht genutzt werden dürfen. Sonderzeichen müssen mit einem \ maskiert werden. Nach dem Namen kommen getrennt durch ein Komma die Tags der Messung. Tags werden indexiert und dazu genutzt um Messwerte zu durchsuchen. Tags werden als Key Value Paar angegeben. Hier sollen Metadaten wie zum Beispiel der Standort des Sensors oder der Name des Servers einzutragen werden zu dem die Datenpunkt/e gehören. Die eigentlichen Werte sind mit einem Leerzeichen von den Tags abgegrenzt und bestehen aus durch Kommas getrennte Key Value Feldern. Der letzte Wert einer Zeile ist der

---

<sup>10</sup>vgl. solid-IT-gmbh (2022)

<sup>11</sup>vgl. Influxdata (2022)



**Listing 2:** Wetterdaten CSV

```

STATIONS_ID;MESS_DATUM; QN;PP_10;TT_10;TM5_10;RF_10;TD_10;eor
73;202205120000; 2; -999; 12.9; 11.2; 84.2; 10.3;eor
73;202205120010; 2; -999; 12.7; 11.2; 84.9; 10.2;eor
73;202205120020; 2; -999; 12.9; 11.4; 83.0; 10.1;eor
73;202205120030; 2; -999; 12.4; 10.7; 86.9; 10.3;eor
73;202205120040; 2; -999; 12.4; 10.5; 86.2; 10.2;eor
73;202205120050; 2; -999; 12.3; 10.3; 85.5; 9.9;eor
73;202205120100; 2; -999; 12.1; 10.1; 88.1; 10.2;eor
73;202205120110; 2; -999; 11.7; 9.9; 90.1; 10.1;eor
73;202205120120; 2; -999; 11.7; 10.0; 89.0; 10.0;eor
73;202205120130; 2; -999; 11.9; 10.2; 86.3; 9.7;eor
73;202205120140; 2; -999; 12.3; 10.6; 83.5; 9.6;eor
73;202205120150; 2; -999; 12.4; 10.9; 83.3; 9.7;eor
73;202205120200; 2; -999; 11.7; 9.8; 86.2; 9.5;eor
73;202205120210; 2; -999; 11.6; 9.6; 88.5; 9.8;eor
73;202205120220; 2; -999; 11.4; 9.4; 88.6; 9.6;eor
73;202205120230; 2; -999; 11.8; 9.9; 88.7; 10.0;eor
73;202205120240; 2; -999; 11.4; 9.9; 88.7; 9.6;eor
73;202205120250; 2; -999; 11.5; 9.7; 89.5; 9.8;eor
73;202205120300; 2; -999; 11.6; 10.0; 88.4; 9.8;eor
73;202205120310; 2; -999; 11.4; 10.3; 87.5; 9.4;eor
73;202205120320; 2; -999; 11.6; 9.9; 89.0; 9.9;eor
73;202205120330; 2; -999; 12.1; 10.4; 87.3; 10.1;eor
73;202205120340; 2; -999; 12.1; 10.6; 87.2; 10.0;eor
73;202205120350; 2; -999; 11.9; 10.2; 87.2; 9.8;eor
...

```

In der CSV Datei gibt es 9 Felder.

STATION_ID	Gibt an von welcher Station die Werte stammen
MESS_DATUM	Gibt an wann gemessen wurde im Format "%Y%m%d%H%M" Also Jahr Monat Tag Stunde Minute als eine zusammengeschriebene Zahl.
QN	Gibt die Qualität der Messwerte an. Hier gibt es die Werte 1 bis 3 <ol style="list-style-type: none"> <li>1. nur formale Kontrolle bei Dekodierung und Import</li> <li>2. Kontrolle mit individuell festgelegten Kriterien</li> <li>3. ROUTINE automatische Kontrolle und Korrektur mit QUALIMET</li> </ol>
PP_10	Luftdruck auf Stationshöhe
TT_10	Lufttemperatur auf 2 Meter höhe
TM5_10	Lufttemperatur auf 5cm höhe
TD_10	relative Luftfeuchtigkeit auf 2m höhe
eor	END OF RECORD kann ignriert werden.

In dieser CSV Datei sind die Daten mit einem Semikoln voneinander getrennt. Der erste Wert in der CSV Datei ist die STATIONS\_ID auf die später noch weiter eingegangen

wird. Danach folgt das Feld Mess\_Datum Formatiert nach dem

### **3.5.2 Wetterdaten abrufen**

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

### **3.5.3 Wetterdaten verarbeiten**



## 4 Zusammenfassung

Dieses Dokument ist eine Hilfe, um die Formalien für eine Bachelor-Thesis an der FHDW bei der Verwendung von  $\LaTeX$  zu erfüllen und dabei möglichst viele Automatismen von  $\LaTeX$  zu nutzen. Eine Absprache mit dem betreuenden Professor ist dennoch ratsam.

## Anhang

### Anhangsverzeichnis

Anhang 1: Gesprächsnotizen . . . . .	12
Anhang 1.1: Gespräch mit Werner Müller . . . . .	12

## **Anhang 1    Gesprächsnotizen**

### **Anhang 1.1    Gespräch mit Werner Müller**

Gespräch mit Werner Müller am 01.01.2013 zum Thema XXX:

- Über das gute Wetter gesprochen
- Die Regenwahrscheinlichkeit liegt immer bei ca. 3%
- Das Unternehmen ist total super
- Hier könnte eine wichtige Gesprächsnotiz stehen

## Quellenverzeichnis

### Internetquellen

- ComputerWeekly.de, Redaktion (2021). *Definition Zeitreihendatenbank (Time Series Database, TSDB)*. URL: <https://datascience.eu/wiki/what-the-heck-is-time-series-data-and-why-do-i-need-a-time-series-database/> (besucht am 21. Mai 2021).
- Data-Science-Team (2020). *What the heck is time-series data (and why do I need a time-series database)?* URL: <https://www.computerweekly.com/de/definition/Zeitreihendatenbank-Time-Series-Database-TSDB> (besucht am 9. Mai 2022).
- Dix, Paul (2021). *Why Time Series Matters for Metrics, Real-Time Analytics and Sensor Data*. URL: <http://get.influxdata.com/rs/972-GDU-533/images/why%20time%20series.pdf> (besucht am 10. Mai 2022).
- Fangman, Sam (2019). *The Time Has Come for a New Type of Database*. URL: <https://medium.datadriveninvestor.com/the-time-has-come-for-a-new-type-of-database-47cf8df1667a> (besucht am 10. Mai 2022).
- Hajek Vlasta Pour Ales, Kudibal Ivan (2019). *Why Time Series Matters for Metrics, Real-Time Analytics and Sensor Data*. URL: <http://get.influxdata.com/rs/972-GDU-533/images/why%20time%20series.pdf> (besucht am 27. Mai 2022).
- Influxdata (2021). *Why You Should Migrate from SQL to NoSQL for Time Series Data*. URL: <https://www.influxdata.com/from-sql-to-nosql/> (besucht am 20. Mai 2022).
- Influxdata (2022). *API Quick Start*. URL: [https://docs.influxdata.com/influxdb/v2.2/api-guide/api\\_intro/](https://docs.influxdata.com/influxdb/v2.2/api-guide/api_intro/) (besucht am 21. Mai 2022).
- solid-IT-gmbh (2022). *DB-Engines Ranking*. URL: <https://db-engines.com/de/ranking> (besucht am 10. Mai 2022).

## Ehrenwörtliche Erklärung

Hiermit erkläre ich, dass ich die vorliegende Studienarbeit - Entwurf selbständig angefertigt habe. Es wurden nur die in der Arbeit ausdrücklich benannten Quellen und Hilfsmittel benutzt. Wörtlich oder sinngemäß übernommenes Gedankengut habe ich als solches kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Paderborn, 29. Mai 2022

---

Henrik Mertens